Pizza Ordering App - JavaFX

# 1. Project Overview

This is a JavaFX-based Pizza Ordering Application designed to allow users to manage pizza orders efficiently. Users can add, update, delete, and view pizza orders. The application integrates JavaFX for the frontend and MySQL for backend data storage.

# 2. Folder Structure

The project follows the standard Maven directory structure:  
.  
├── pom.xml  
├── src  
│ ├── main  
│ │ ├── java  
│ │ │ └── com  
│ │ │ └── example  
│ │ │ └── pizza  
│ │ │ ├── dao  
│ │ │ ├── model  
│ │ │ ├── util  
│ │ │ ├── MainApp.java  
│ │ │ ├── PizzaController.java  
│ │ │ └── SecondaryController.java  
│ │ └── resources  
│ │ └── com  
│ │ └── example  
│ │ ├── pizza\_order.fxml  
│ │ ├── primary.fxml  
│ │ └── secondary.fxml  
│ └── test  
│ └── java  
│ └── com  
│ └── example  
│ └── pizza  
│ └── util  
│ └── BillCalculatorTest.java

# 3. Features

- Add, View, Update, and Delete pizza orders.  
- Input fields for customer name, mobile number, pizza size, and toppings.  
- TableView to display all orders with dynamic updates.  
- Data validation and clean UI/UX with JavaFX.  
- Database integration using JDBC with MySQL.

# 4. Requirements

- Java 17+  
- JavaFX 17.0.2  
- MySQL Server  
- Maven 3.6+  
- IDE: VS Code or IntelliJ IDEA

# 5. Setup Instructions

1. Clone the repository to your local machine.  
2. Open the project in your preferred IDE.  
3. Configure JavaFX SDK in the build path.  
4. Make sure MySQL is running and create the database and required tables.  
5. Update your DB credentials in the DAO class.  
6. Run using: `mvn clean javafx:run`

# 6. Screenshots

Screenshot 1: Application Interface

A screenshot of a computer

AI-generated content may be incorrect.

Screenshot 2: Data Added

A screenshot of a computer

AI-generated content may be incorrect.

Screenshot 3: TableView Display

A screenshot of a computer

AI-generated content may be incorrect.

## 7. Disclaimer

This project was developed by Saddam Hussain Safi for academic purposes. ChatGPT was used for troubleshooting, guidance, and partial logic validation. All original implementation and documentation was completed independently.